This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/MIC.2020.3019920, IEEE Internet Computing

The Internet of Bodies Needs a Human Data Model

Niko Mäkitalo[1], Daniel Flores Martin[2], Javier Berrocal[2], José García-Alonso[2], Petri Ihantola[1], Aleksandr Ometov[3], Juan Manuel Murillo[2], and Tommi Mikkonen[1]

[1]University of Helsinki, Helsinki, Finland
{niko.makitalo, petri.ihantola, tommi.mikkonen}@helsinki.fi

[2]University of Extremadura, Cceres, Spain
{dfloresm, jberolm, jgaralo, juanmamu}@unex.es

[3]Tampere University, Tampere, Finland
aleksandr.ometov@tuni.fi

Abstract—Today, the number of interconnected devices and the amount of personal information gathered by them increases tremendously resulting in the need for development tools to harness its potential. New devices are continually being introduced in the daily life of people, and they are already producing an unprecedented amount of data related to people's well-being. However, taking advantage of such information to create innovative Internet of Bodies solutions heavily relies on manually gathering the needed information from several sources on services and the devices involved. In this paper, we present a novel Human Data Model – a new tool to combine personal information from several sources, perform computations over that information, and proactively schedule computer-human interactions. Developers that use the proposed model would obtain an opportunity to create the Internet of Bodies solutions using high-level abstractions of the users' personal information and taking advantage of the distributed approach of the model.

Index Terms—Internet of Things, Internet of Bodies, Internet of Wearable Things, Human Data Model, Human-Computer Interaction, Internet of Bodies, IoT, IoB

1 Introduction

Presently, an increasing number of interoperating systems produces a flood of sensitive personal and intimate data. In the cyber world, human data is generated by our actions in social media and various types of cloud services. In the physical world, the data is now generated by the ever-growing amount of data pumps (wearables, sensors, mobile devices, etc.) next to us, on the network edges, where we and our devices are located. In essence, these devices connect the human body to the Internet, paving the way to the Internet of Bodies (IoB) paradigm, a network of human bodies whose integrity and functionality rely at least in part on the Internet and related technologies [1].

Unveiled by IoB, the provided human data has plenty of potential for both completely new types of software applications as well as for improving conventional applications. Particularly important, said data is considered as data collectively gathered from our surroundings being directly related to our well-being from a smart healthcare perspective [2, 3]. However, utilizing this data with present approaches may be a complicated task. Its' access and usage requires that an IoT device sends the data to a smartphone or a backend service. This setting has serious flaws, such as the Internet companies swamp personal data gathered from their customers is a privacy nightmare, and transferring the data to and from the cloud is intolerable for latency-sensitive systems [4]. It can only be fixed by reconsidering the way the data is utilized, calling for coordination between the Edge, the Fog, and the Cloud [5].

In this article, we present a new programming model called Human Data Model (HDM) and API for accessing and interacting with it. It is an abstract model for collecting data concerning our lives and then processing the data into an actionable form. With the model, it is also possible to build applications that proactively schedule computer-human interactions to allow the computing infrastructure to take the initiative to serve the users [6] better. At the end of this article, we study how developing IoB applications with HDM is improved in contrast to mobile-app-based and cloud-based approaches.

2 Internet of Bodies – State-of-the-Art

The popularity of mobile and web apps have lead the way for digitization and define how people embrace technology. [7]. While there are established approaches for mobile

1089-7801 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: La Trobe University. Downloaded on September 21,2020 at 05:11:54 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/MIC.2020.3019920, IEEE Internet Computing

and web app development, there currently is no such approaches for developing IoB solutions – IoB solutions are primarily developed either by implementing mobile apps or by implementing web-based services. It originates from the fact that the current wearable IoB devices are connected either to other devices (typically a smartphone-gateway) or directly to a Cloud service. This leads to people being connected and producing an important data flow [8] that must be managed by applications to make the most of it and help people in their day-to-day lives.

Programming IoB applications is commonly made by leveraging APIs offered by frameworks or by providers, such as Suunto Cloud API (https://apizone.suunto.com) or Wellness Warehouse Engine (https://w2e.fi). The development of the IoT has brought us an instantly updated lifestyle and has encouraged the creation of a myriad of applications and services to meet the needs of consumers. The exposure of the services' digital assets via APIs in a controlled manner is known as *API economy*. API economy is fast, convenient, and allows everyone from the smallest developers to the largest companies to add interaction capabilities based on web APIs.

Using Healthcare APIs, developers can easily integrate their healthcare applications with complex ecosystems that capture, store, and process data about people's health, e.g., Apple's HealthKit with ResearchKit and CareKit provide an excellent opportunity to conduct studies and to build apps that leverage exciting health and wellness data. However, this approach leaves plenty of room for technological improvements. Firstly, the processing and collecting are separated, and data needs to be transferred. Secondly, the approach does not provide solutions for connecting and leveraging various services for computations or leveraging the data provided by these services. Thirdly, it is only limited to Apple's supported devices, which also has benefits since the devices are homogenized.

In addition to complex ecosystems, the integration of these APIs can also be done with intelligent devices, that allow a versatile placement and provide flexibility and comfort to monitor people health status. Many of these devices can be worn or placed on clothing, on or under the skin, and on almost any part of the body: wrist, ankle, waist, chest, arm, legs, etc. These devices can detect a plethora of different variables, the number of steps, activity performed, e.g., speed, distance, stairs climbed, etc. Also, other alternatives include the use of gloves, rings or necklaces, among others, to monitor more delicate parameters such as oxygen saturation and blood glucose level, or heart rate. There are numerous studies devoted entirely to the analysis of these types of sensors for monitoring human activity such as [9] and [10]. The work [11] studies different types of sensors that can be used for monitoring health status from the biomedical-technological perspective. Based on [11], an extract of different IoB sensors and the possible clinical use are illustrated in on top in Figure 1. Sheth et. al work on Computing for Human Experience [12] has been inspiring our work for Human Data Model.

SIDEBAR: DEMAND FOR NEW HEALTHCARE AND WELL-BEING SERVICES

One of the unavailable reality aspects worldwide is related to the population aging. According to the United Nations, 1 in 11 people in the world are over the age of 65, and 1 in 6 will be in the year 2050. This trend is even more acute in some rural European regions. Regions like Extremadura in Spain or Alentejo in Portugal combine population aging with low population density as well as with migration to more developed regions creating demographic deserts in developed countries.

This reality requires innovative solutions. *The International Institute for Research and Innovation on Ageing (4IE)* [13] develops its activity in the listed cross-border regions with the most aged rural areas in Europe are located – 24% of the population are aged over 65. This project aims to bring Human Data Model based solutions to these areas to help in providing health-related services. To achieve this goal, some challenges that are usually not in the focus of technological solutions should be addressed. Specifically, challenges regarding the lack of technological infrastructure, like a stable Internet connection, the lack of technical skills of the population, or the economic cost of health-related services deployment must be addressed.

3 SCENARIO: NEW HEALTHCARE AND WELL-BEING SERVICES FOR ELDERLY PEOPLE

Although most of the elderly are not "digital natives", supporting their everyday life with the help of IoB technology is becoming a priority in modern society (as discussed in the sidebar). To illustrate our point, let us consider the following scenario.

John, 72 years old, has different wearables: shoes with sensors that measure the number of steps and type of physical activity, a smart bracelet that measures his heart rate, an implant that allows him to know his blood glucose level at all times, and even a shirt with sensors that can detect different blood pressure levels to report possible shocks. John's technological wrap does not represent any barrier in his everyday life since devices are tiny, well-integrated, and easy to maintain. In any case, potential inconveniences would be compensated by the benefit provided by those internet services monitoring important John's vital signs through the data provided by the sensors.

In addition, John is active in social media, providing the information about his hobbies, concerns, and interests. Processing this information allows to discover and anticipate *sensations* – or data abstractions – that are relevant to him. Gathering all the information recorded about John (both coming from sensors and his activity on the Internet) enables to finally detect more complex sensations and adapt the behavior of technology in his surroundings to his needs. Thus, John's everyday activities , like going for a walk, exercising, or taking his medication, are registered. That allows to create highly personalized model – a model that

1089-7801 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: La Trobe University. Downloaded on September 21,2020 at 05:11:54 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/MIC.2020.3019920, IEEE Internet Computing

can be used to encourage a more active daily life by asking different friends in social media to perform some related activity such as going for a walk or even creating virtual gamified experiences at a particular time of the day.

The sensations are stored locally on John's devices and synchronized continuously among them in his wearable cloud. In addition, some data may be accessible to third parties when connected to the Internet. Hence, the sensations can be shared and synchronized with family, friends, or health care experts so that, e.g., John can be proactively advised to visit a doctor in case necessary. Moreover, introducing new smart health monitoring devices in John's surrounding is a smooth process since either John's existing devices automatically detect when a new data source is available, and then John simply grants access to this information, or downloads an app on his smartphone that takes care of accessing the data provided by the new source device or service.

Naturally, assuring John's sensitive data privacy is a priority. For that, he can decide what information he shares and with whom. In addition, such information is handled and stored locally so that no personal data is sent to third-party servers to be processed without John's consent.

4 Human Data Model for the IoB

The scenario where IoB and other devices and services work as seamlessly as described above requires solutions that go beyond the current state of the art development approaches. In this paper, we introduce HDM and its realization (see Figure 1), which collects the data related to our digital and physical lives, and then refines this data into more abstract sensations. The model offers an API for accessing and interacting with the now more meaningful data, but the abstractions can also be used for proactively scheduling computer-human interactions.

4.1 Perception and interaction

An interaction can be considered as a process that consists of sensation (input from the world), anticipation (what events are expected), adaptation (how to react to unforeseen events), and action (output to the world) [14]. Altogether, these give a meaning for the interaction. Perception, in turn, is essential for the interaction, and there are many flavors of perception – the perception of persons, of utility, and of information, to name some of them. These are all essential from an interaction perspective.

Therefore, perception is a crucial HDM quality. It helps to generate more meaningful sensations by collecting raw data from the physical and cyber worlds. The motivation behind is to allow applications running on the devices around the user's current environment to anticipate the relevant sensations, and then to adapt the device behavior so that the environment can serve the user in a more personalized way. As social relationships are also fundamental in how entities interact, we have defined how the relationships from the so-called "social world" can be reflected in the sensations.

In the presented scenario, HDM allowed us to detect the *sensations* that took place in John's daily life, e.g., to help anticipating possible health problems and to help discovering routines that lead to healthier and better life quality. An example sensation in the above scenario could be describing that John's heart rate was a bit high in contrast to his typical heart rate after a heavy lunch while he was walking with his friend Paul from the grocery store to the cafeteria. Such a sensation is intuitive and helps John to understand better his health condition and what kind of habits and choices may affect his health in a very concrete way.

4.2 Enabling human–data interaction

Users and devices are different in their needs. Hence, not all of their devices can percept and process all types of data. For instance, the most common IoB devices at the moment can be hard (or impossible) to program. Instead, these devices are typically connected to mobile devices or services and provide APIs for accessing the data. While many approaches aim at homogenizing and providing APIs to the data collected with IoB, the problem yet is that this information may be hard to utilize in practice. One of the critical challenges with wearable technology seems to be usability and user experience. Imagine how hard it could be for an elderly person to start interacting with a smartwatch, for this reason, one essential quality of HDM is that it can improve the interaction with various IoB and other types of devices, like on a mobile phone, cloud service, or even on a web page and provide access to the up-to-date data for easy-to-use applications.

As an example, imagine that when John goes to a doctor's office. He will log in to a SPA (single page web application), and this would create a new HDM instance where all John's medical data would start synchronizing. John would then grant the SPA access to the data he feels comfortable sharing. When the doctor would finish inspecting the data, the HDM instance would die, and all John's data would vanish from that SPA and services it is using.

4.3 Adaptive behavior and growing new instances

Today, it is incredible how much data persons' activities continuously produce. It would be impossible to gather, transfer, or store all this data to one location, and it would not even make sense to do so – people continuously move from one location to another, and the results of the processing often needs to be used on the network edges, where people and devices are located [4]. Therefore, one HDM quality is that it can run on Edge devices, which requires that the device can get a seed object – a data structure containing an identifier for the device and other identifiers related to its owner. The seed object has an essential purpose of linking the physical, cyber, and social worlds entities and processes. The process of deploying the data structure on a device with computing capabilities is so-called "planting a seed," as after getting the seed, the device initializes the model, which then begins to "grow," that is, percept data from physical, cyber, and social worlds.

Not all Edge devices are able to percept or process all types of data. Hence, the quality also allows modifying an instance dynamically and flexibly with processing capabilities at runtime. As depicted in the bottom right corner of Fig. 1, the developers are expected to define new types of sensations with *Sensation generators* continuously – methods that analyze and combine data from various sources

1089-7801 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: La Trobe University. Downloaded on September 21,2020 at 05:11:54 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/MIC.2020.3019920, IEEE Internet Computing

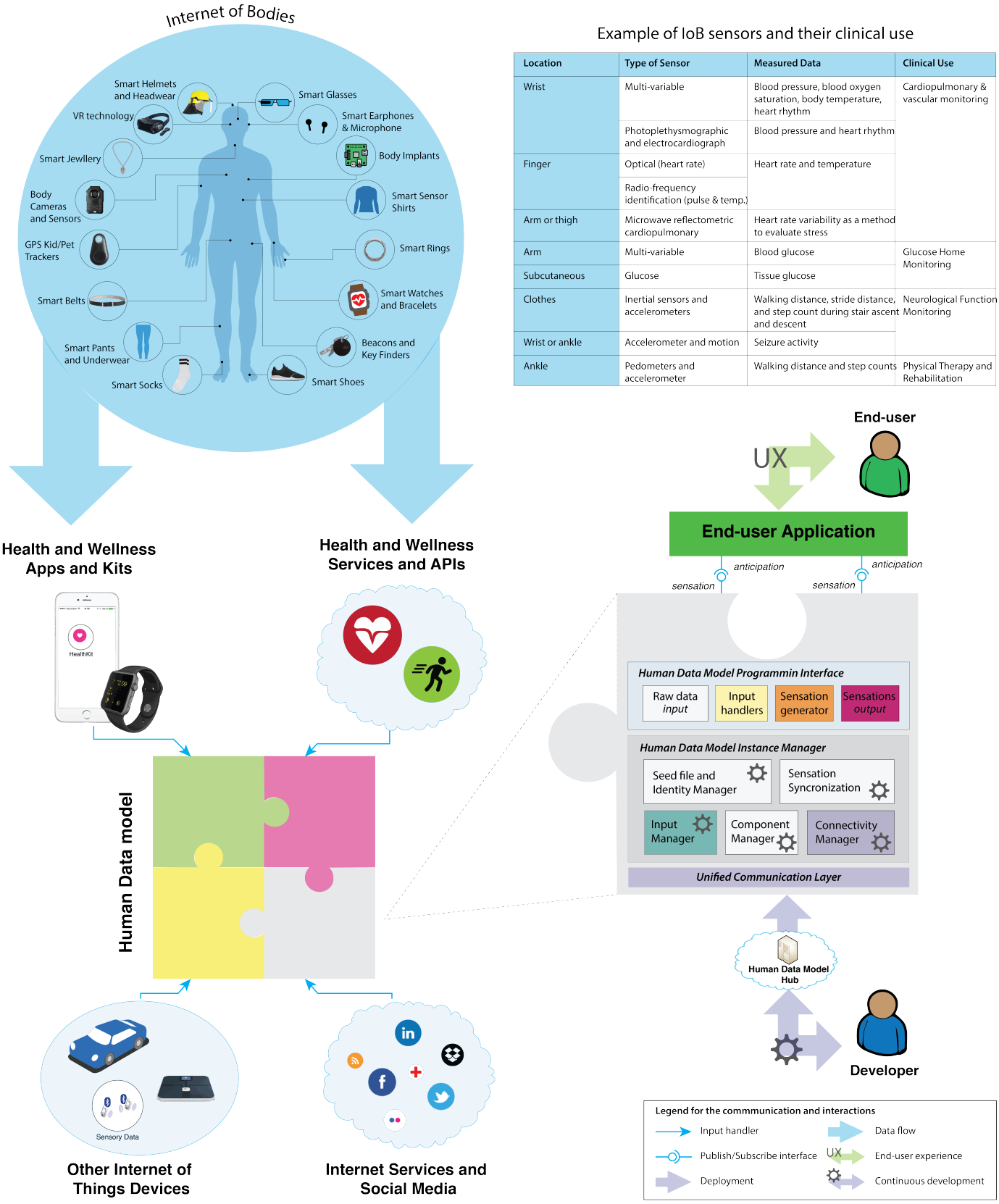
Internet of Bodies
Smart Helmets and Headwear
Smart Glasses
VR technology
Smart Earphones & Microphone
Smart Jewllery
Body Implants
Body Cameras and Sensors
Smart Sensor Shirts
GPS Kid/Pet Trackers
Smart Rings
Smart Belts
Smart Watches and Bracelets
Smart Pants and Underwear
Beacons and Key Finders
Smart Socks
Smart Shoes
Example of IoB sensors and their clinical use
Location | Type of Sensor | Measured Data | Clinical Use
Wrist | Multi-variable | Blood pressure, blood oxygen saturation, body temperature, heart rhythm | Cardiopulmonary & vascular monitoring
Photoplethysmographic and electrocardiograph | Blood pressure and heart rhythm
Finger | Optical (heart rate) | Heart rate and temperature
Radio-frequency identification (pulse & temp.)
Arm or thigh | Microwave reflectometric cardiopulmonary | Heart rate variability as a method to evaluate stress
Arm | Multi-variable | Blood glucose | Glucose Home Monitoring
Subcutaneous | Glucose | Tissue glucose
Clothes | Inertial sensors and accelerometers | Walking distance, stride distance, and step count during stair ascent and descent | Neurological Function Monitoring
Wrist or ankle | Accelerometer and motion | Seizure activity
Ankle | Pedometers and accelerometer | Walking distance and step counts | Physical Therapy and Rehabilitation
Health and Wellness Apps and Kits
Health and Wellness Services and APIs
HealthKit
Human Data model
Sensory Data
Other Internet of Things Devices
Internet Services and Social Media
End-user
UX
End-user Application
anticipation
sensation
anticipation
sensation
Human Data Model Programmin Interface
Raw data input
Input handlers
Sensation generator
Sensations output
Human Data Model Instance Manager
Seed file and Identity Manager
Sensation Syncronization
Input Manager
Component Manager
Connectivity Manager
Unified Communication Layer
Human Data Model Hub
Developer
Legend for the commmunication and interactions
Input handler
Data flow
Publish/Subscribe interface
UX
End-user experience
Deployment
Continuous development

Fig. 1. Human Data Model realization operating with Internet of Bodies sensors and data.

and existing sensations. Correspondingly, the developers define how this data is used by that model instance when it receives the data as well as how the data should be preprocessed or refined before relaying it further by implementing new *input handler* methods. Finally, developers implement *Dispatchers* – a specific type of apps or processes

1089-7801 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: La Trobe University. Downloaded on September 21,2020 at 05:11:54 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/MIC.2020.3019920, IEEE Internet Computing

– that can access data sources that have not been previously supported.

4.4 Ephemeral sensations and integrity

In the real world, the entities around the user may continuously move, and hence the need to interact can be over within a short period. To protect the developers from implementing poorly working interactions and false scheduling, we designed HDM to have a quality of ephemeral sensations, which means that the sensation generated and stored by the model become invalid within a relatively short period – time that can be defined by the developer. Next, the sensations are also being removed from the instances to protect the user privacy.

Referring back to the scenario above, if John takes off his smart shirt or any other wearable device, HDM removes sensations that are based on the data provided by that device in order to prevent relaying false or outdated information to the apps.

5 Implementation

Building on the assumption that the heterogeneity of the IoB entities included in applications is vast, we designed and implemented the framework so that it is dynamic, flexible, and capable of running on as many device platforms as possible. The framework–see the right hand side on Figure 1– is implemented with JavaScript, so the device running an instance needs to be able to execute it. Hence, the runtime environment of HDM's *Instance Manager* can be a web browser, Node.js server, or a JavaScript framework (e.g., iOS JavaScriptCore). JavaScript has quickly become the new de facto language for implementing prototypes and IoT software. However, our motivation for choosing JavaScript as a base for implementation for Human Data Model is as following. First, JavaScript is universal in the sense that it runs on almost any type of device. Second, JavaScript is dynamic and, thus, provides the flexibility to extend the language for our purposes. Third, JavaScript is relatively lightweight but yet powerful enough for the desired HDM-related computations. Finally, JavaScript – while used like with web pages – allows always deploying the latest version of the HDM framework to the executing entities, which allows agile development of the model. *Component Manager* allows deploying input handlers and sensation generators to running instances and keeping them up-to-date if an instance is connected to the Internet.

Seed file and Identity Manager fetches the seed file from the HDM Hub (a central repository for sharing the programming concepts) with HTTP protocol. The purpose of this file is to provide identifiers that connect the social, physical, and cyber worlds. For instance, the seed file typically contains Bluetooth Low Energy (BLE) universally unique identifiers (UUIDs) that represent specific physical devices. This file also may contain information about the user in the cyber and social worlds, like Facebook ID, to help retrieve information about social relationships. Such publicly available information from Internet can be distributed to all other HDM instances.

Connectivity Manager supports different connectivity types and enables other (possibly more constrained) devices to send data to the Edge device and hence to the instances. The used protocol and connectivity type have a substantial effect on how the user experiences the interactions. Both local and global communication approaches have their advantages and disadvantages: Local, like the BLE, typically offer stable and fast coordination, but their support among the entities is limited. Global, on the other hand, is supported by the most IoT devices, but the communication may go via services, and coordination may be sensitive to the quality of the Internet connection. However, based on our studies, BLE and Socket.IO fit well for interaction coordination purposes, and the key is to support them both with a communication abstraction – as we do with the *Unified Communication Layer* [5]. It typically leads to extended star topologies, where some devices with both local and global connectivity act as gateways for the ones that only support local communication.

Connectivity Manager is also responsible for the security side of the system operation. In contrast to conventional infrastructure-like links, wearable devices tend to operate in a peer-to-peer manner, and the connectivity to the Cloud may not always be present. Therefore, combined information security solutions are utilized in order to enable the operation for cases of both available and intermittent connectivity to the Internet. Simultaneously, both security and communications sides should be operating taken into consideration the natural limitations of most of the wearable devices, i.e., resource constraints, namely, computational and battery ones. Proper selection of the technology may improve not only the user experience but also enable reliable, secure and private data processing.

6 Discussion

Implementation of the IoB applications requires developing certain features or data collection, synchronization, combining it with data from other sources, and carrying out the necessary processing to make the most out of it. Traditionally, the heaviest processes are deployed on specialized servers or in the Cloud, requiring a permanent Internet connection. Table 1 shows the most critical tasks for carrying out the entire development process of IoB applications by comparing HDM to traditional mobile-app-based and Cloud-based approaches from capturing the data on people to their subsequent processing, and further so that the processed information can be used by the applications to anticipate people's needs.

As the table indicates, HDM has several advantages. All the information processing can be done locally on wearables, taking advantage of their excellent computing capacity, or globally, as computations can be carried out in the Cloud when needed by considering the available resources in the environment. Another added advantage is that the Internet connection does not have to be permanent. As a trade-off, edge devices' computing capacity cannot be compared to the Cloud servers', so processing takes longer. Also, adaptability is one of the advantages of the proposed model, since new devices can be added to the ecosystem, and data can be easily obtained and associated with discovering new sensations to improve the behavior of IoB applications.

1089-7801 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: La Trobe University. Downloaded on September 21,2020 at 05:11:54 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/MIC.2020.3019920, IEEE Internet Computing

TABLE 1
Comparing mobile-app-based (M) and cloud-service-based (C) approaches to Human Data Model (H) for Internet of Bodies application development.

Task	Approach	Comments and questions	Support
Collecting data	M	• Typically data is transferred to applications or separate storage (e.g., HealthKit) • Several app-specific implementations may be required to cover all the needed data sources	Internet not required, platform-specific implementations often needed
	C	• Data is transferred to remote servers over the Internet • May consume a lot of device resources, specially bandwidth and battery • Transferring can be slow or become a bottleneck	Internet required for transferring data
	H	• Same Dispatchers are used for accessing data and feeding it for several different types of applications • Same Input handlers can be used for adding new input data types and their preprocessing • Optimal solution can be chosen: data may be transferred to the Cloud or stored on the mobile device	Internet required only for accessing data stored in Cloud services.
Synchronizing data	M	• All raw data typically synchronized via the Internet • Data synchronization routines between the devices must be implemented by the developers	Internet typically required for synchronizing data
	C	• All data is synchronized via the Cloud which takes a lot of communication resources • As services and number of users scale, the cost can be increased considerably	Internet required for synchronizing all data
	H	• High abstraction level sensations are automatically synchronized between authorized devices running HDM instances • Raw data may also be synchronized but is not recommended	Internet required only for synchronizing between possible HDM Cloud instances
Combining data	M	• Application developers may implement their own solutions for combining data from various sources	No clear programming concepts for combining the data
	C	• Application developers may implement their own solutions for combining data from various sources	No clear programming concepts for combining the data
	H	• Supports combining data from various sources with Input handlers and Sensation generators	Clear programming concepts for combining data to meaningful sensations
Detecting sensations	M	• Detecting changes in local data is typically implemented with event handlers • Different device platforms require implementing different solutions	Local interface for detecting changes in data
	C	• Can provide global publish/subscribe interface for detecting changes • May be hard to detect changes in the data on other services or on devices • May be slow – the detected change in the data may come too late	Global publish/subscribe interface often provided for detecting changes in data
	H	• Provides publish/subscribe interface for detecting changes in the data • Changes can easily be detected on the mobile devices and on services	Publish/subscribe interface for detecting sensations on all users devices and services, locally and globally
Decentralizing computation and cooperation	M	• All the computations take place on the wearable device • Constant computation off all data coming from all sources may drain the mobile device battery • May lead to bottlenecks if the computation requires a lot of CPU time	Developer has very limited control on which device or service the computations take place
	C	• All the computations take place on centralized services • May lead to bottlenecks in communication or computation	Developer has no control over on which device or service the computations take place
	H	• Computations are encouraged to take place near the datas origin • Computations can be deployed e.g., by considering the resources required for transferring data in contrast to the local resources like battery status	Developer has control over on which device or service the computations take place
Deployability of the approach	M	• Applications are deployed via the application stores	No deployments during runtime are supported
	C	• Applications deployed to the servers and are immediately available for all devices	Dynamic deployment possible
	H	• Applications are deployed and updated when the Internet connectivity is available • Automatic and almost real-time deployment	Dynamic deployment supported
Adding new devices and services	M	• New devices may be connected via short-range communication technology (e.g., BLE) • New services can be connected to apps via APIs	Each developer implements his/her own solutions for adding new data sources
	C	• The service must provide support for the desired new device • The service must provide support for the desired new service	Adding new data sources may be cumbersome
	H	• Provides support for adding new devices (with the ability to grow from seed) • New services can be added by implementing new Dispatchers	Adding new devices and services is easy and one of the key goals

TABLE 2
Human Data Model framework consumption estimation.

Data required in the framework				Consumption estimation	
Device	**Description**	**Operation**	**Data size (Bytes)**	**Battery (mAh)**	**Data traffic (bytes)**
Shoes	Steps counter and activity detection	getInfo(ID_shoes)	216	25,09	1,21 Kb
Heart rate bracelet	Heart rate	getInfo(ID_bracelet)	34	3,95	189,91
Glucose level implant	The glucose level in the blood	getInfo(ID_implant)	40	4,65	223,43
Blood pressure t-shirt	Blood pressure	getInfo(ID_tshirt)	43	4,99	240,19
-	John profile (JP)	post(JP)	327	42,37	2,77 Kb

1089-7801 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: La Trobe University. Downloaded on September 21,2020 at 05:11:54 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/MIC.2020.3019920, IEEE Internet Computing

Finally, the resource consumption of creating the HDM has being estimated using the framework described in [15]. It allows obtaining the data and energy consumption of a device by defining the primitives of a concrete use case, considering *get* and *post* primitives over different resources. In this case, the evaluation is conducted over the John's case study. The required information is specified in Table 2. This information is stored in plain text, considered as a common practice on the device side. Moreover, the table includes the consumption estimation performed in the framework.

7 Conclusions

More and more Internet of Bodies devices are being developed with the consequent generation of vast amount of the entailing data. Due to that, new tools are required to simplify the gathering, relating, and processing of this information to perceive the peoples' environment and act to anticipate possible problems, moods, or needs accordingly. In this work, a model has been presented allowing one to capture the personal information related coming from heterogeneous sources for its later processing and promoting proactive interactions between intelligent devices and people. We demonstrated how abstract sensations can be generated with Human Data Model and how these can be used in the software applications. We believe that such facilities are essential for the Internet of Bodies as well as other related approaches where data related to individuals is processed.

Visit GitHub to access Human Data Model implementation and documentation for developers: *https://humandatamodel.github.io/*.

8 Acknowledgements

This work was supported by 4IE+ project (0499_4IE_PLUS_4_E) funded by the Interreg V-A España-Portugal (POCTEP) 2014-2020 program, by the project RTI2018-094591-B-I00 (MCIU/AEI/FEDER, UE) and FPU17/02251 grant, by project IB18030 funded by the Government of Extremadura. The work of N. Mäkitalo was supported by the Academy of Finland (project 328729). We would especially want to thank Professor Yevgeni Koucheryavy from Tampere University for his contributions and insight.

References

[1] A. M. Matwyshyn, "The Internet of Bodies," *William & Mary Law Review*, vol. 61, no. 1, 2019.

[2] A. Solanas, C. Patsakis, M. Conti, I. S. Vlachos, V. Ramos, F. Falcone, O. Postolache, P. A. Pérez-Martínez, R. Di Pietro, D. N. Perrea *et al.*, "Smart Health: A Context-Aware Health Paradigm within Smart Cities," *IEEE Communications Magazine*, vol. 52, no. 8, pp. 74–81, 2014.

[3] G. D. Abowd, "Beyond Weiser: From Ubiquitous to Collective Computing," *Computer*, no. 1, pp. 17–23, 2016.

[4] A. V. Dastjerdi and R. Buyya, "Fog Computing: Helping the Internet of Things Realize Its Potential," *Computer*, vol. 49, no. 8, pp. 112–116, 2016.

[5] N. Mäkitalo, A. Ometov, J. Kannisto, S. Andreev, Y. Koucheryavy, and T. Mikkonen, "Safe, Secure Executions at the Network Edge: Coordinating Cloud, Edge, and Fog Computing," *IEEE Software*, vol. 35, no. 1, pp. 30–37, 2018.

[6] J. Miranda, N. Mäkitalo, J. Garcia-Alonso, J. Berrocal, T. Mikkonen, C. Canal, and J. M. Murillo, "From the Internet of Things to the Internet of People," *Internet Computing, IEEE*, vol. 19, no. 2, pp. 40–47, Mar 2015.

[7] Y. Liu, X. Liu, Y. Ma, Y. Liu, Z. Zheng, G. Huang, and M. B. Blake, "Characterizing restful web services usage on smartphones: A tale of native apps and web apps," in *2015 IEEE International Conference on Web Services*. IEEE, 2015, pp. 337–344.

[8] W. Tan, M. B. Blake, I. Saleh, and S. Dustdar, "Social-network-sourced big data analytics," *IEEE Internet Computing*, vol. 17, no. 5, pp. 62–69, 2013.

[9] A. Pantelopoulos and N. G. Bourbakis, "A Survey on Wearable Sensor-based Systems for Health Monitoring and Prognosis," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 40, no. 1, pp. 1–12, 2009.

[10] J. Kim, A. S. Campbell, B. E.-F. de Ávila, and J. Wang, "Wearable Biosensors for Healthcare Monitoring," *Nature biotechnology*, vol. 37, no. 4, pp. 389–406, 2019.

[11] G. Appelboom, E. Camacho, M. E. Abraham, S. S. Bruce, E. L. Dumont, B. E. Zacharia, R. DAmico, J. Slomian, J. Y. Reginster, O. Bruyère *et al.*, "Smart Wearable Body Sensors for Patient Self-Assessment and Monitoring," *Archives of public health*, vol. 72, no. 1, p. 28, 2014.

[12] A. Sheth, P. Anantharam, and C. Henson, "Physical-cyber-social computing: An early 21st century approach," *IEEE Intelligent Systems*, vol. 28, no. 1, pp. 78–82, 2013.

[13] J. Garcia-Alonso, J. Berrocal, J. M. Murillo, D. Mendes, C. Fonseca, and M. Lopes, "Situational-context for virtually modeling the elderly," in *Ambient Intelligence – Software and Applications –, 9th International Symposium on Ambient Intelligence*, P. Novais, J. J. Jung, G. Villarrubia González, A. Fernández-Caballero, E. Navarro, P. González, D. Carneiro, A. Pinto, A. T. Campbell, and D. Durães, Eds. Cham: Springer International Publishing, 2019, pp. 298–305.

[14] U. Neisser, *Cognition and reality: Principles and implications of cognitive psychology.* WH Freeman/Times Books/Henry Holt & Co, 1976.

[15] J. Berrocal, J. Garcia-Alonso, C. Vicente-Chicote, J. Hernández, T. Mikkonen, C. Canal, and J. M. Murillo, "Early Analysis of Resource Consumption Patterns in Mobile Applications," *Pervasive and Mobile Computing*, vol. 35, pp. 32–50, 2017.

Niko Mäkitalo is a Postdoctoral researcher at the University of Helsinki, Department of Computer Science. He received Ph.D. in Computer Science in 2016. Niko's main interests are Web technologies in the context of Fog Computing, IoT programming, and Robotics. He is Associate Editor of IEEE Software Blog and a member of ACM and IEEE Computer Society. Contact him at niko.makitalo@helsinki.fi

1089-7801 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: La Trobe University. Downloaded on September 21,2020 at 05:11:54 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/MIC.2020.3019920, IEEE Internet Computing

Daniel Flores-Martin is a Ph.D student at the University of Extremadura (Spain). He is currently working at the Computing and Telematics Systems Department of the University of Extremadura. His research interests are mobile computing, context-awareness, crowd sensing and Internet of Things. Flores-Martin received the MSc in Computer Science Engineering in 2019. Contact him at dfloresm@unex.es

Javier Berrocal is an associate professor in the Department of Informatics and Telematics System Engineering at the University of Extremadura (Spain) and co-founder of the Startup Gloin. He received a Ph.D. degree (with European Mention) in 2014. His research interests include mobile computing, context awareness, pervasive systems, the Internet of Things, and fog computing. Contact him at jberolm@unex.es

José García-Alonso is an associate professor in the Department of Informatics and Telematics System Engineering at the University of Extremadura (Spain) and co-founder of the Startups Gloin and Viable. He received a Ph.D. degree (with European Mention) in 2014. His main research interests are eHealthCare, eldercare, Mobile Computing, Context-Awareness and Pervasive Systems. Contact him at jgaralo@unex.es

Petri Ihantola is an associate professor of big data learning analytics at the University of Helsinki, Finland. Ihantolas research interests span across educational data mining and building educational software with a particular focus on smart content, automated assessment, and learning analytics in computing education. Contact him at petri.ihantola@helsinki.fi

Aleksandr Ometov is a Postdoctoral Researcher at Tampere University, Finland. He is currently focused on H2020 ITN/EJD A-WEAR project. He received his Dr.Sc. (Tech.) in 2018. His major research interests are wireless communications, information security, heterogeneous networking, cooperative communications, wearable and blockchain applications. Contact him at aleksandr.ometov@tuni.fi

Juan M. Murillo is a full professor of software engineering at the University of Extremadura and co-founder of the Startups Gloin and Viable. His research interests include software architectures, mobile computing and cloud computing. Contact him at juanmamu@unex.es

Tommi Mikkonen is a Professor of Software Engineering at the University of Helsinki, Finland. Tommis research focuses on software architectures, agile methodologies, web technologies, and connected devices. He has published over two hundred peer-reviewed conference and journal papers. Tommi received his Ph.D. in information technology in 1999. Contact him at tommi.mikkonen@helsinki.fi

1089-7801 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: La Trobe University. Downloaded on September 21,2020 at 05:11:54 UTC from IEEE Xplore. Restrictions apply.